Evaluation of the Gramotron Parser for German

Franz Beil[1], **Detlef Prescher**[2], **Helmut Schmid**[3], **Sabine Schulte im Walde**[3]

[1]TEMIS, Rue de Ponthieu 59, 75008 Paris, France
franz.beil@temis-group.com

[2]DFKI GmbH, Language Technology Lab, Stuhlsatzenhausweg 3, 66123 Saarbrücken, Germany
prescher@dfki.de

[3]IMS, Universität Stuttgart, Azenbergstr. 12, 70174 Stuttgart, Germany.
{Helmut.Schmid,Sabine.Schulte.im.Walde}@IMS.Uni-Stuttgart.DE

Abstract

The paper describes an experiment in inside-outside estimation of a lexicalized probabilistic context free grammar for German. Grammar and formalism features which make the experiment feasible are described. Successive models are evaluated on precision and recall of phrase markup consisting of labels for noun chunks and subcategorization frames. Our approach to parsing is a blend of symbolic and stochastic methods where we use evaluation results in both incremental grammar development and validation of selected output to be used in lexical semantic clustering. Our results are that (i) scrambling-style free phrase order, case morphology, subcategorization, and NP-internal gender, number and case agreement can be dealt within a lexicalized probabilistic context-free grammar formalism, and (ii) inside-outside estimation appears to be beneficial, however relies on a carefully built grammar and an evaluation based on carefully selected linguistic criteria. Additionally, we report experiments on overtraining with inside-outside estimation, especially focusing on comparison of the results of mathematical and linguistic evaluations.

1. Introduction

From 1997 to 2000, the Gramotron group of the Institute for Natural Language Processing at Stuttgart University developed a stochastic parser for German (Beil et al. (1999), Schulte im Walde et al. (2001)). The symbolic component of the final parsing system is a manually written context-free grammar consisting of several thousand head-marked rules. Its stochastic component consists of probability weights assigned to the lexicalised grammar rules and to the lexical choice events by the so-called inside-outside algorithm (Lari and Young, 1990), the standard procedure for unsupervised training of a stochastic context-free grammar parsing free text. For training and parsing, the implementations of Carroll (1997b) and Schmid (1999a) were used.

The Gramotron parsing system was designed to be used for the induction of a semantically annotated lexicon of German nouns and verbs (Rooth et al., 1999). Accordingly, the grammar development focus was on the recognition of the grammatical relations between nouns and verbs.

Furthermore, since the parsing results were an intermediate step in an experiment to learn a semantic lexicon, reliable parsing results had to be acquired rapidly. We decided for an incremental grammar development, thus minimizing grammar development efforts in the early project phase.

The context-free grammar for German was developed in three stages: for (i) verb-final clauses, (ii) relative clauses, and (iii) verb-first and verb-second clauses. In this paper, we describe a concluded experiment and evaluation of the parsing system covering constructions (i) and (ii).

Grammar development and stochastic training was controlled by two types of evaluation: (i) an information-theoretic evaluation based on perplexity values measured on training and test corpora of free text, and (ii) a linguistic evaluation of noun chunks with case features and verb frame recognition on a manually annotated test corpus.

2. Data

The data for our experiments are two sub-corpora extracted from a 200 million token newspaper corpus, (a) a sub-corpus containing 450,000 verb-final clauses with a total of 4 million words, and (b) a sub-corpus containing 1,1 million relative clauses with a total of 10 million words. Apart from non-finite clauses as verbal arguments, there are no further clausal embeddings, and the clauses do not contain any punctuation except for a terminal period. The average clause length is 9.16 and 9.12 words per clause, respectively.

We used a finite-state morphological analyser (Schiller and Stöckert, 1995) to assign multiple morphological features such as part-of-speech tag, case, gender and number to the corpus words, partly collapsed to reduce the number of analyses. For example, the word *Bleibe* (either the case ambiguous feminine singular noun 'residence' or a person and mode ambiguous finite singular present tense verb form of 'stay') is analysed as follows:

```
analyse> Bleibe
1. Bleibe+NN.Fem.Akk.Sg
2. Bleibe+NN.Fem.Dat.Sg
3. Bleibe+NN.Fem.Gen.Sg
4. Bleibe+NN.Fem.Nom.Sg
5. *bleiben+V.1.Sg.Pres.Ind
6. *bleiben+V.1.Sg.Pres.Konj
7. *bleiben+V.3.Sg.Pres.Konj
```

Reducing the ambiguous categories leaves the two morphological analyses

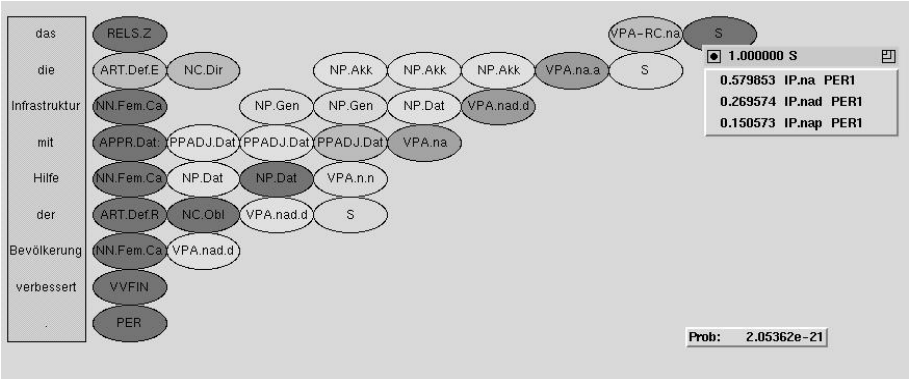

Figure 1: Chart Browser for Grammar Development

```
Bleibe { NN.Fem.Cas.Sg, VVFIN }
```

Apart from assigning morphological analyses the tool in addition serves as lemmatiser (cf. (Schulze, 1996)).

3. The German Context-Free Grammar

The context-free grammar consists of 5,033 rules with lexical head markings. With very few exceptions (rules for coordination, S-rule), the rules do not have more than two daughters. The 220 terminal categories in the grammar correspond to the collapsed corpus tags assigned by the morphology.

Grammar development is facilitated by (a) grammar development environment of the feature-based grammar formalism YAP (Schmid, 1999b), and (b) a chart browser that permits a quick and efficient discovery of grammar bugs (Carroll, 1997a). Figure 1 shows that the ambiguity in the chart is quite considerable even though grammar and corpus are restricted.

The grammar covers 92.43% of the verb-final and 91.70% of the relative clauses, i.e. the respective part of the corpora are assigned parses.

In the following, we describe two essential parts of the grammar, the noun chunks and the definition of subcategorisation frames. For details concerning prepositional phrases, adjectival chunks, adverbial chunks, complex determiners, and the treatment of coordination see (Schulte im Walde, 2000).

3.1. Noun Chunks (NCs)

On nominal categories, in addition to the four cases `Nom`, `Gen`, `Dat`, and `Akk`, case features with a disjunctive interpretation (such as `Dir` for `Nom` or `Akk`) are used. The grammar is written in such a way that non-disjunctive features are introduced high up in the tree. Figure 2 illustrates the use of disjunctive features in noun projections: the terminal `NN` contains the four-way ambiguous `Cas` case feature; the N-bar (`NN1`) and noun chunk `NC` projections disambiguate to two-way ambiguous case features `Dir` and `Obl`; the weak/strong (`Sw/St`) feature of `NN1` allows or prevents combination with a determiner, respectively; only at the noun phrase `NP` projection level, the case feature appears in disambiguated form. The use of disjunctive case features results in some reduction in the size of the parse forest. Essentially the full range of agreement inside the noun phrase is enforced. Agreement between the subject `NP` and the tensed verb is not enforced by the grammar, in order to control the number of parameters and rules.

The noun chunk definition refers to Abney's chunk grammar organisation (Abney, 1996): the noun chunk (`NC`) is a projection that excludes post-head complements and (adverbial) adjuncts introduced higher than pre-head modifiers and determiners, but includes participial pre-modifiers with their complements.

3.2. Subcategorisation Frames

The grammar distinguishes four subcategorisation frame classes: active (`VPA`), passive (`VPP`), non-finite (`VPI`) frames, and copula constructions (`VPK`). A frame may have maximally three arguments. Possible arguments in the frames are nominative (n), dative (d) and accusative (a) NPs, reflexive pronouns (r), PPs (p), and non-finite VPs (i). The grammar does not distinguish plain non-finite VPs from *zu*-non-finite VPs. The grammar is designed to distinguish between PPs representing a verbal complement or adjunct: only complements are referred to by the frame type. The number and the types of frames in the different frame classes are given in Table 1.

Frame Class	#	Frame Types

`VPA`	16	`n, na, nd, np, nad, nap, ndp`
		`ni, di, nai, ndi`
		`nr, nar, ndr, npr, nir`

`VPP`	18	`n, np-s, d, dp-s, p, pp-s`
		`nd, ndp-s, np, npp-s, dp, dpp-s`
		`i, ip-s, ni, nip-s, di, dip-s`

`VPI`	8	`-, a, d, p, r, ad, ap, dp, pr`
`VPK`	2	`n, i`

Table 1: Subcategorisation Frame Types

German, being a language with comparatively free phrase order, allows for scrambling of arguments. Scrambling is reflected in the particular sequence in which the arguments of the verb frame are saturated. Compare Figure 3 as example of a canonical subject-object order within an active transitive frame *der sie liebt* 'who loves her' and its scrambled object-subject order *den sie liebt* 'whom she loves'.

Abstracting from the active and passive realisation of

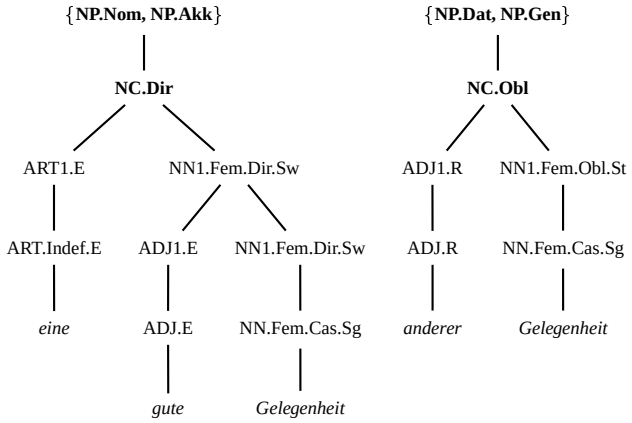

Figure 2: Noun Projections

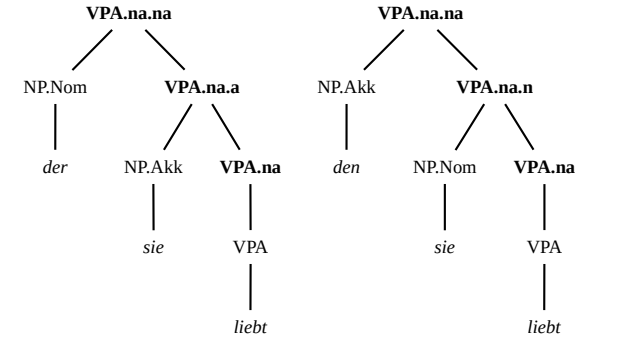

Figure 3: Realising Scrambling Effect in the Grammar Rules

an identical underlying deep-level syntax we generalise over the alternation by defining a top-level subcategorisation frame type, e.g. `IP.nad` for `VPA.nad`, `VPP.nd` and `VPP.ndp-s` (with `p-s` a prepositional phrase within passive frame types representing the deep-structure subject, realisable only by PPs headed by *von* or *durch* 'by'); see Figure 4 for an example, presenting the relative clauses *der die Frau verfolgt* 'who follows the woman', *die verfolgt wird* 'who is followed' and *die von dem Mann verfolgt wird* 'who is followed by the man'.

4. Probability Model

The probabilistic grammars are parsed with `LoPar`[1] (Schmid, 1999a), a head-lexicalised probabilistic context-free parser. The parser is an implementation of the Left-Corner algorithm for parsing and of the Inside-Outside algorithm for parameter estimation. Probabilistic context-free parsing (Lari and Young, 1990) maps a CFG to a probability model by assigning a probability to each grammar rule.

Innovative features of LoPar are head lexicalisation, lemmatisation, parameter pooling, and a sophisticated smoothing technique.

[1]LoPar is basically a re-implementation of the Galacsy tools which were developed by Glenn Carroll in the SFB, but LoPar provides additional functionality.

Syntactically, a head-lexicalised probabilistic context-free grammar (HPCFG) (Carroll, 1995; Carroll and Rooth, 1998) is a PCFG in which one of the right hand side categories of each grammar rule is marked as the head of the projection. The lexical head of a terminal category is the respective word form. Thus, lexical head properties, i.e. words, are propagated through head chains.

HPCFGs assign the following probability[2] to a parse tree `T`:

$$
\begin{aligned}
P(T) \;=\;& P_{start}(\text{cat}(\text{root}(T))) \cdot \\
& P_{start}(\text{head}(\text{root}(T))|\text{cat}(\text{root}(T))) \cdot \\
& \prod_{n \in T} P_{rule}(\text{rule}(n)|\text{cat}(n), \text{head}(n)) \cdot \\
& \quad n \text{ : non-terminal} \\
& \prod_{n \in T} P_{choice}(\text{head}(n)|\text{cat}(n), \text{cat}(\text{p}(n)), \text{head}(\text{p}(n))) \cdot \\
& \quad n \neq root(T) \\
& \prod_{n \in T} P_{rule}(\texttt{<terminal>}|\text{cat}(n), \text{head}(n)) \cdot \\
& \quad n \text{ : terminal} \\
& \prod_{n \in T} P_{lex}(\text{word}(n)|\text{cat}(n), \text{head}(n)) \\
& \quad n \text{ : terminal}
\end{aligned}
$$

Five families of probability distributions are relevant here. $P_{start}(C)$ is the probability that C is the category of the root node of a parse tree. $P_{start}(h|C)$ is the probability that a root node of category C bears the lexical head h. $P_{rule}(r|C,h)$ is the probability that a node of category C with lexical head h is expanded by rule r. $P_{choice}(h|C,C_p,h_p)$ is the probability that a (non-head) node of category C has the lexical head h given that the parent category is C_p and the parent head is h_p. $P_{rule}(\texttt{<terminal>}|C,h)$ is the probability that a node of category C with lexical head h is a terminal node. $P_{lex}(w|C,h)$, finally, is the probability that a terminal node with category C and lexical head h expands to the word form w.

In order to reduce the prohibitively large number of lexical parameters that have to be estimated, we employed linguistic generalisations for parameter reduction: lemmatisation and parameter pooling. Using uninflected lemma rather than inflected word form for lexicalisation eliminates splitting of estimated frequencies among inflectional forms. Parameter pooling is based on the assumption that lexical choice probabilities are unlikely to depend on inflectional features like gender, case, number etc. of categories or argument order in verb frames. For instance, there are (at least) nine different inflectional patterns for projecting the adjective *alt* (old) and *Buch* (book) to an `NN1` category. Instead of assigning a lexical choice probability

$$P_{choice}(alt|\texttt{ADJ.w}_\texttt{i}, \texttt{NN1.x}_\texttt{i}\texttt{.y}_\texttt{i}\texttt{.z}_\texttt{i}, Buch)$$

[2]The auxiliary functions `cat`, `head`, `p(arent)`, `word` and `rule` return the syntactic category, the lexical head, the parent node, the dominated word or the expanding grammar rule of a node. `root` returns the root node of a parse tree and `<terminal>` is a constant.

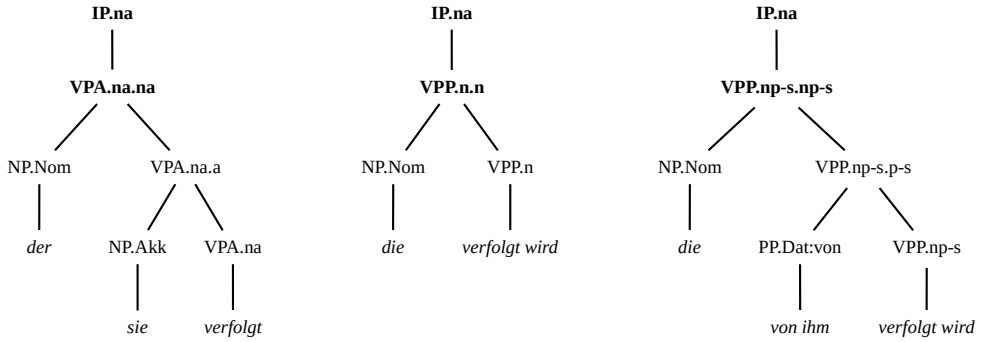

Figure 4: Generalising over the Active-Passive Alternation of Subcategorisation Frames

for each possible combination of w, x, y, z, the combinations are pooled to a single distribution

$$P_{choice}(alt|\texttt{ADJ}, \texttt{NN1}, \text{Buch})$$

for all inflectional variations of `NN1 -> ADJ NN1`. We obtain a single probability distribution for adjectival modifiers. In result, frequent observation of *altes Buch* in the trainng data also increases the probability of *alter Bücher*. For argument filling into verb frames, categories of the form `VP.x.y` are pooled to `VP.x` and active, passive and non-finite verb frames are pooled according to shared arguments, disregarding the saturation state of the frame. For instance, P_{choice} of a particular noun is the same as accusative NP head in the transitive active frame or nominative NP head in the passive frame of a particular verb (*[dass] sie den Hund füttert 'she feeds the dog'*, *der Hund gefüttert wird 'the dog is fed'*).

5. Grammar Training

5.1. Training Strategy

The training in our main experiment was performed in the following steps:

1. Initialisation of all CFG rules with identical frequencies. (Comparative initialisations with random frequencies had no effect on the model development.)

2. Unlexicalised training: The training corpus was parsed once, re-estimating the frequencies twice.

3. Lexicalisation: The unlexicalised model was turned into a lexicalised model by (i) setting the probabilities of the lexicalised rule probabilities to the values of the respective unlexicalised probabilities and (ii) initialising the lexical choice and lexicalised start probabilities uniformly.

4. Lexicalised training:
   Three training iterations were performed on the training corpus, re-estimating the frequencies after each iteration.

For training the model parameters we used 90% of the corpora, a total of 1.4 million clauses. The remaining 10% of serve as heldout data to measure overtraining.

Our experiments have shown that training an unlexicalised model first improves overall results. The optimal training strategy proceeds with few parameter re-estimations of an unlexicalised model. Without re-estimations or with a large number of re-estimations the model was effected to its disadvantage. With less unlexicalised training more changes during lexicalised training take place later on.

Comparative numbers of iterations (up to 40 iterations) in lexicalised training showed that more iterations did not have any further effect on the model.

6. Evaluation

Our evaluation methods were chosen to monitor the development of the grammar, to control the grammar training, and compare different training regimes. As part of our larger project of lexical semantic clustering, the parsing system had the specific task to collect corpus frequencies for pairs of a verbal head and its subcategorisation frame and frequencies for the nominal fillers of slots in a subcategorisation frame. The linguistic evaluation focuses on the reliability of these parsing results.

6.1. Mathematical evaluation

A		B		C	
1:	52.0199	1:	53.7654	1:	49.8165
2:	25.3652	2:	26.3184	2:	23.1008
3:	24.5905	3:	25.5035	3:	22.4479
⋮	⋮	⋮	⋮	⋮	⋮
15:	24.2861	57:	25.0549	90:	22.1443
16:	24.2861	58:	25.0549	95:	22.1443
17:	24.2867	59:	25.055	96:	22.1444

Table 2: Overtraining (iteration: cross-entropy on heldout data)

In order to control the amount of unlexicalised training, we measured overtraining by comparing the perplexity of the model on training and heldout data (or, respectively, cross-entropy[3] on heldout data in the experiments

[3]For a corpus consisting of sentences of a certain average length (`avg`), one can easily transform these cross-entropy values (`cross`) to the better known values of word perplexity (perp)

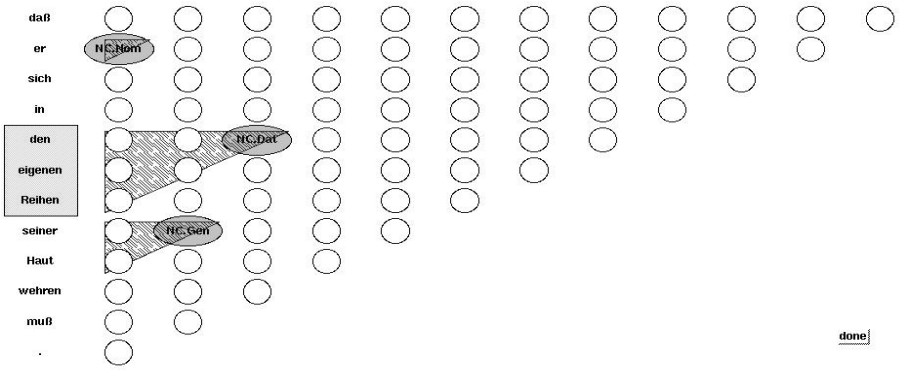

Figure 5: Chart Browser for manual constituent markup

in (Beil et al., 1999)). While perplexity on training data is theoretically guaranteed to converge through subsequent iterations, increasing perplexity on heldout data indicates overtraining. Table 2 shows comparisons of different sizes of training and heldout data (training/heldout) for unlexicalised training in an older experiment (Beil et al., 1999): (A) 50k/50k, (B) 500k/500k, (C) 4.1M/500k. The overtraining effect is indicated by the increase in cross-entropy from the penultimate to the ultimate iteration in the tables.

In previous experiments (Beil et al., 1999), we compared in more detail the mathematical evaluation with the linguistic evaluation of precision/recall measures on categories of different complexity through iterative unlexicalised training. The comparison shows that the mathematical criterion of overtraining may lead to bad results from a linguistic point of view. While precision/recall measures for low-level structures such as NCs converge, iterative unlexicalised training up to the overtraining threshold is disadvantageous for the evaluation of complex categories like subcategorisation frames. We observed precision/recall values for verb frames settling even below the results with a randomly initialised grammar. So the mathematical evaluation can only serve as a rough indicator whether the model reaches towards an optimum, but linguistic evaluation determines the optimum.

6.2. Linguistic evaluation

Although an appropriate treebank is available for German (the NEGRA treebank, cf. Skut et al. (1997) for an overview), we did not use it for our evaluation. One reason for this is the restriction of our initial grammar development to verb final and relative clauses while the treebank, of course, annotates full clauses. It turned out to be difficult to extract respective sub-treebanks. On the other hand, we did not intend to carry out the standard parser evaluation method of measuring precision/recall on phrase boundaries and crossing brackets (the PARSEVAL scheme) for which treebanks are widely used. Bracketing information is rather uninteresting for our objectives and we reckoned that rich structures as generated by our grammar would likely punished by the crossing bracket measure. (For a more general overview of problems using the crossing brackets measure for parser evaluation see (Carroll et al., 1998).)

Moreover, in transforming our bracketing to treebank annotation standards, we feared to loose too much information deemed important for our evaluation. In our efforts to find a transformation that maps treebank structures to a selection of ours (noun and verb chunks), we found two mapping problems: (i) mapping treebank phrase spans to our chunk spans and (ii) finding an information-preserving mapping from our labels to treebank labels. Concerning the first, it turned out to be difficult to define noun chunk ends within treebank NPs. An even harder problem is finding the rich information in our verbal category labels (i.e. type and frame annotation) in treebank VPs.

So we decided to build our own test data: Rather than pursuing the efforts of finding an appropriate treebank-to-gramotron transformation, we performed detailed evaluations of individual frames and of a set of selected verbs.

Test data The linguistic parameters of the models were evaluated concerning the identification of NCs and subcategorisation frames. We randomly extracted 200 relative clauses and 200 verb-final clauses from the test data and hand-annotated the relative clauses with noun chunk labels, and all of the clauses with frame labels. In addition, we extracted 100 randomly chosen relative clauses for each of the six verbs *beteiligen* 'participate', *erhalten* 'receive', *folgen* 'follow', *verbieten* 'forbid', *versprechen* 'promise', *versuchen* 'try', and hand-annotated them with their subcategorisation frames. The particular selection of verbs aims to be representative for the variety of verb frames defined in our grammar.

The manual annotation was facilitated by use of a chart browser. The labellers filled the appropriate chart cells with category names by selecting category labels from a given list that is displayed on clicking a cell. Figure 5 gives an example of NC-labelling which visualises the determination of NC-ranges via cell selection. Frames are annotated as IP

using the formula

$$\texttt{perp} = 10^{\,\texttt{avg}^{-1}\cdot\texttt{cross}}$$

(assuming that the cross-entropy is computed by a logarithm based on 10). For example, an average lenghth of avg=9.2 and a cross-entropy of cross=24.2 yields a word perplexity perp=427.0, which is a value comparable to the values presented in Schulte im Walde et al. (2001).

labels, i.e. they are always in the same chart cell and frame ranges are trivial.

Best-first consistency Our linguistic evaluation of the probability models is a version of measuring best-first consistency (Briscoe and Carroll, 1993). We made the models determine the Viterbi parses (i.e. maximum probability parses) of the test data and extracted the categories of interest (i.e. noun chunks and subcategorisation frame types). Only the relevant categories but not the entire Viterbi parses were compared with the annotated data. NCs were evaluated according to (i) range and (ii) range and label, i.e. category name. The subcategorisation frames were evaluated according to the frame label only. Precision and recall measures are defined as follows:

$$precision = \frac{correct}{guesses} \qquad recall = \frac{correct}{baseline}$$

with *baseline* referring to the set of annotated categories in the test corpus, *guesses* referring to the set of range/label annotated categories identified in Viterbi parses, and *correct* counting the cases where the chunk/label identified by the parser is a match to the annotator's choice ($correct = guesses \cap baseline$).

Overall results The precision values of the "best" model according to the training strategy were as in Table 3.

Noun Chunks		Subcategorisation Frames on Sub-Corpora	
range	range+label	relative clauses	verb final clauses
98%	92%	63%	73%

Subcategorisation Frames on Specific Verbs					
beteiligen 'participate'	*erhalten* 'receive'	*folgen* 'follow'	*verbieten* 'forbid'	*versprechen* 'promise'	*versuchen* 'try'
48%	61%	88%	59%	80%	49%

Table 3: Precision Values on Noun Chunks and Subcategorisation Frames

For comparison reasons, we evaluated the subcategorisation frames of 200 relative clauses extracted from the training data. Interestingly, there were no striking differences concerning the precision values.

Evaluation of training regimes Figure 6 present the strongly different development of noun chunk and subcategorisation frame representations within the models, ranging from the untrained model until the fifth iteration of lexicalised training. NCs were modelled sufficiently by an unlexicalised trained grammar. Unexpectedly, lexicalisation impaired the modelling slighlty. This observation is supported by related experiments of German noun chunking on an unrestricted text corpus (Schmid and Schulte im Walde, 2000). It remains to be explored whether the number of low-frequent nominal heads is—despite the use of lemmatisation for parameter reduction—still prohibitively large because of the pervasive morpho-syntactic process of noun compounding in German.

Verb phrases in general needed a combination of unlexicalised and lexicalised training, but the representation strongly depended on the specific item. Unlexicalised training advanced frequent phenomena (compare, for example, the representation of the transitive frame with direct object for *erfahren* and with indirect object for *folgen*), lexicalisation and lexicalised training improved the lexicalised properties of the verbs, as expected.

Parameter pooling Regarding the frame evaluation, we also did a test on the effects of parameter pooling in lexicalised traininng. Without pooling of frame categories the precision values for low-frequent phenomena such as non-finite frame recognition was significantly lower, e.g. the precision for the verb *versuchen* was 9% less than with pooling. This result suggests investigations into the importance of training data size and research into other pooling possibilities.

6.3. Error Analysis

A detailed investigation of frame recognition showed the following interesting feature developments:

- Highly common subcategorisation types such as the transitive frame are learned in unlexicalised training and then slightly unlearned in lexicalised training. Less common subcategorisation types such as the demand for an indirect object are unlearned in unlexicalised training, but improved during lexicalised training.
- It is difficult and was not effectively learned to distinguish between prepositional phrases as verbal complements and adjuncts.
- The active present perfect verb complexes and passive of condition were confused, because both are composed by a past participle and a form of *to be*, e.g. *geschwommen ist* 'has swum' vs. *gebunden ist* 'is bound'.
- Copula constructions and passive of condition were confused, again because both may be composed by a past participle and a form of *to be*, e.g. *verboten ist* 'is forbidden' vs. *erfahren ist* 'is experienced'.
- Noun chunks belonging to a subcategorised non-finite clause were partly analysed main verb arguments. For instance, *der ihn zu überreden versucht* 'who him$_{acc}$ tried to persuade' was parsed as demanding an accusative plus a non-finite clause instead of recognising that the accusative object is subcategorised by the embedded infinitival verb.
- Reflexive pronouns may trigger either a reflexive or, by virtue of projecting to an accusative or dative noun chunk, a transitive frame. The correct or wrong choice of frame type containing the reflexive pronoun was learned consequently right or wrong for different verbs. For instance, the verb *sich befinden* 'to be situated' was generally parsed as a transitive, not as inherently reflexive.

6.4. Shortcomings and evaluation alternatives

We are aware that there are some desirable aspects missing from our evaluation.

Firstly, we did not evaluate the relations between lexical heads directly, the main task our parsing system was designed for. Subcategorisation frame and noun chunk label

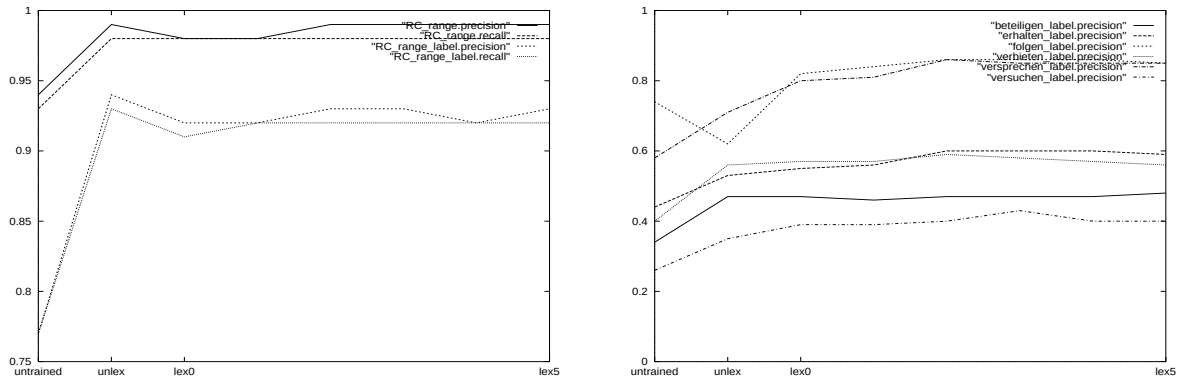

Figure 6: Development of Precision and Recall Values on Noun Chunk Range and Label (left-hand side), and Precision Values on Subcategorisation Frames for Specific Verbs (right-hand side)

recognition serve only as indirect evidence of how well our model does on recognising scrambling of verbal arguments. Because noun chunk annotation is not confined to verb argument slots—PP embedded noun chunks were annotated as well—and a detailed error analysis on noun chunk labels is missing, it remains unclear whether scrambled nominal arguments are subject to more errors than the remarkable 92% precision on NC labels suggests. Similarly, correctly recognised verb frames with a prepositional argument have not been evaluated as to whether the assigned PP argument is actually the correct one.

Secondly, we did not evaluate the correctness of lexical heads of phrases.

Relevant evaluation schemes that capture our shortcomings are the evaluation of dependency structure as described in (Lin, 1995) or the proposal of evaluating of grammatical relations of Carroll et al. (1998). Both evaluation proposals address the importance of selectively evaluating parsing systems with respect to specific types of syntactic phenomena rather than measuring overall performance as in "traditional" evaluation schemes. Selective evaluation is a definite desideratum for our own evaluation task. The proposals also point to a way to automatically extract evaluation relevant relations from an annotated corpus. Inquiring about the feasibility of mapping Negra, the treebank for German, to a respective test corpus will hopefully provide a more comprehensive basis for our future evaluations of head–head relations.

7. Conclusion

Our approach to parsing is a combination of symbolic and stochastic methods. The symbolic component usually involves a very high degree of overgeneration leaving disambiguation to the stochastic component. To facilitate disambiguation by statistical means, the symbolic component relies on certain categorial generalizations and uses non-standard categories to reduce the parameter space or allow for parameter pooling. We used evaluation results in both incremental grammar development and validation of selected output to be used in lexical semantic clustering.

Our principal result is that scrambling-style free-er phrase order, case morphology and subcategorization, and NP-internal gender, number and case agreement can be dealt with in a head-lexicalized PFCG formalism. A second result is that inside-outside estimation appears to be beneficial, however relies on a carefully built grammar where parses can be evaluated by carefully selected linguistic criteria.

Furthermore, we reported experiments on overtraining with inside-outside estimation. These experiments are made possible by the carefully built grammar and our evaluation tools, especially allowing to compare and to relate the results of our mathematical and linguistic evaluation. In combination, these provide a general framework for investigating training regimes for lexicalized PCFGs.

However, there are two relevant aspects missing from our evaluation. First, we did not evaluate grammatical relations directly. Frame and NC case recognition give only a crude idea of how well our model does on recognizing e.g. scrambled subject and direct object. Because NC evaluation is not confined to verb argument slots, the picture is distorted. Second, we did not evaluate the correctness of lexical heads of phrases. Clearly, if we can overcome our difficulties to map Negra, the treebank for German, to a respective test corpus, a more valuable basis for future evaluations of head–head relations supplied by the gramotron parsing system is provided.

Finally, although there is no guarantee that the maximization of the likelihood of the training data (which the inside-outside algorithm performs) also improves the linguistic correctness of the resulting syntactic analyses, our experiments show that in practice this is the case. Gaining more insight into the relationship between linguistic plausibility and likelihood of linguistic analyses will be an interesting future research topic.

8. References

Steven Abney. 1996. Chunk stylebook. Technical report, SfS, Universität Tübingen.

Franz Beil, Glenn Carroll, Detlef Prescher, Stefan Riezler, and Mats Rooth. 1999. Inside-outside estimation of a lexicalized PCFG for German. In *Proceeding of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-99)*, College Park, Maryland.

Ted Briscoe and John Carroll. 1993. Generalised prob-

abilistic LR parsing for unification-based grammars. *Computational Linguistics*, 19(1):25–60.

Glenn Carroll and Mats Rooth. 1998. Valence induction with a head-lexicalized PCFG. In *Proceedings of EMNLP-3*, Granada.

John Carroll, Ted Briscoe, and Antonio Sanfilippo. 1998. Parser evaluation: a survey and a new proposal. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, Granada, Spain.

Glenn Carroll. 1995. *Learning Probabilistic Grammars for Language Modeling*. Ph.D. thesis, Department of Computer Science, Brown University.

Glenn Carroll, 1997a. *Manual pages for* `charge`, `hyparCharge`. IMS, Universität Stuttgart.

Glenn Carroll, 1997b. *Manual pages for* `supar`, `ultra`, `hypar`. IMS, Universität Stuttgart.

K. Lari and S. J. Young. 1990. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*, 4:35–56.

Dekang Lin. 1995. A dependency-based method for evaluating broad-coverage parsers. In *IJCAI-95*.

M. Rooth, S. Riezler, D. Prescher, G. Carroll, and F. Beil. 1999. Inducing a semantically annotated lexicon via EM-based clustering. In *Proc. of ACL'99*.

Anne Schiller and Chris Stöckert, 1995. *DMOR*. IMS, Universität Stuttgart.

Helmut Schmid and Sabine Schulte im Walde. 2000. Robust German Noun Chunking with a Probabilistic Context-Free Grammar. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING-00)*, pages 726–732, Saarbrücken, Germany, August.

Helmut Schmid, 1999a. *LoPar. Design and Implementation*. Insitut für Maschinelle Sprachverarbeitung, Universität Stuttgart.

Helmut Schmid. 1999b. *YAP: Parsing and Disambiguation with Feature-Based Grammars*. Ph.D. thesis, Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart.

Sabine Schulte im Walde, Helmut Schmid, Mats Rooth, Stefan Riezler, and Detlef Prescher. 2001. Statistical grammar models and lexicon acquisition. In *Linguistic Form and its Computation*. CSLI, Stanford, CA.

Sabine Schulte im Walde. 2000. The German statistical grammar model: Development, training and linguistic exploitation. Arbeitspapiere des Sonderforschungsbereichs 340 *Linguistic Theory and the Foundations of Computational Linguistics* 162, Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart, December.

Bruno Maximilian Schulze, 1996. *GermLem – ein Lemmatisierer für deutsche Textcorpora*. IMS, Universität Stuttgart.

Wojciech Skut, Brigitte Krenn, Thorsten Brants, and Hans Uszkoreit. 1997. An annotation scheme for free word order languages. In *Proceedings of the Fifth Conference on Applied Natural Language Processing ANLP-97*, Washington, DC.